Introduction:

Welcome to BBS Express! Professional. We think you will find this BBS to be the most powerful BBS **ever written** for the Atari 8-bit line of computers.

We know you are anxious to get the board up and running, but please take the time to read through the manual completely before attempting to configure the board. An overall understanding of how the board is set up and how it runs will help you when it comes time to decide where to put things on your drives.

This revision comes to you as a user with something that had been lacking in all previous revisions. A check for command validity. Each command on this disk has been uniquely **CRC'd**, and **DATE SET**, for your specific Node number.

What this means to you is that **you/we** now have the ability to check your commands online. If the file passes the **CRC** check then it will function properly. If you receive a **FAIL** it is possible for the command to function in a manner that it was not designed for; or do weird things that could cause you hours of grief rebuilding the board.

Each **COMMAND** on the **MASTER DISK** has been tested relentlessly and will perform the task for which it was designed properly.

Throughout this manual, we will refer to SpartaDOS's command structure with the understanding that you are familiar with how to use SpartaDOS. We won't attempt to teach you how to use SpartaDOS within the context of this manual. Assuming you have a working knowledge of the DOS, you should have no trouble setting up and running the BBS. If you are a novice SpartaDOS user, keep your SpartaDOS manual handy for quick reference.

We will be maintaining an 'Official' Pro Support board for the exclusive use of registered owners. The phone number is 706-798-2474, or 801-967-8738. You may call in Atasii or Ascii at 300, 1200, 2400 or 9600 baud. After registering on the board, you will be disconnected. You may call back the next day and will have been validated by then. This support board will have placed on it any upgrades to the various modules as well as new modules as they are written.

Thanks again for purchasing BBS Express! Professional. Drop us a line if you would like to see something added to the board that would be of benefit to the majority of SysOp running the system. We will be adding commands and upgrading the program from time to time, so check in with the support board periodically for the latest news on improvements.

System Requirements

BBS Express! PRO's power is made possible by tailoring the program to a specific CPU group and a specific DOS. You should use SpartaDos 3.3a in order to run Express! PRO. As of this writing, the SpartaDos X cartridge can NOT be used with PRO. While PRO will boot under the 4.19 version of SpartaDos X, it does not run reliably. We STRONGLY suggest you stick with 3.2a while Sparta 3.2d will work there are still errors that have been corrected in X33a.Dos. By the very nature of SpartaDOS, you are limited to using an Atari 800XL or 130XE. PRO will not run on an 800 due to the fact that SpartaDOS 3.3a won't run it. Any memory upgrade available for the 800XL or 130XE will work with PRO as long as you install your ramdisk handler before running the BBS program.

If you own the R-time-8 clock cartridge, PRO will take full advantage of it. If you don't own one, you will need to set the SpartaDOS software clock before running PRO. In either case, the TDLINE.COM file should be run prior to running the board. We recommend using a hard drive to run PRO, but it isn't absolutely necessary. PRO's commands are loaded into memory as required, so the slower your storage device, the longer it will take to load in these commands. The best solution, even when running with a hard drive, is to keep the BBS's primary command subdirectory on the ramdisk. This will afford the best response time to the user.

BBS Express! PRO's shell loads into memory at $2C00. This is the portion of the BBS that always remains in memory and has control most of the time. You must load TDLINE.COM from the SpartaDOS disk before PRO will run. If using an 850 or PR:C connection you must load RS232.COM for the 850, and PRC.SYS for the PR:C. These can be included in your STARTUP.BAT file. Optimally, you should have room to load your favorite accessories (such as a ramdisk handler) as long as MEMLO goes no higher than $2C00.

Any Hayes compatible modem can be used with BBS Express! PRO. How Hayes compatible? Well... the modem must be able to answer the phone with an ATA command. Must be able to hang up the phone when the DTR line drops low. Not too much to ask a BBS modem to do. If you have a modem that uses the extended command set, PRO will auto-detect the callers baud rate by detecting the CONNECT, CONNECT 1200 and CONNECT 2400 messages returned by the modem.

With the availability of newer high speed modems be sure you have set your modem to the proper specifications.

What's on The Disk

BBS Express! PRO is shipped to you on 3 disks.

Disk 1 contains:
Front Side

BBS.COM	- the BBS shell module.
SYSEDIT.COM	- the syseditor.
MAKEULOG.COM	- userlog creation program.
MAKEMSGB.COM	- message base creation program.
UEXTEND.COM	- extends the size of the userlog
MAKESUB.BAT	- create subdirectory batch file.
CMDPATH.DAT	- the command path table (user changeable)
DLTIME.DAT	- the download time allowance (user changeable)
DATAPATH.DAT	- the data path table (user changeable)
OVLPATH.DAT	- the overlay drive table (user changeable)
HELP.DAT	- the drives your help files are created on
	* The above data files are placed under your Pro subdirectory your SYSDATA.DAT file is on

Back Side

Data>	- Data files for support programs.
Help40>	- All the atascii and ascii help files for 40 column users.
Help80>	- All the atascii and ascii help files for 80 column users.

Disk 2 & 3 contains all of the system command modules that are required for Pro to run. These modules should be placed in the PRO>COMMANDS>A> subdirectory our hard drive and/or ramdisk. Please note that there are commands on BOTH sides of the disk.

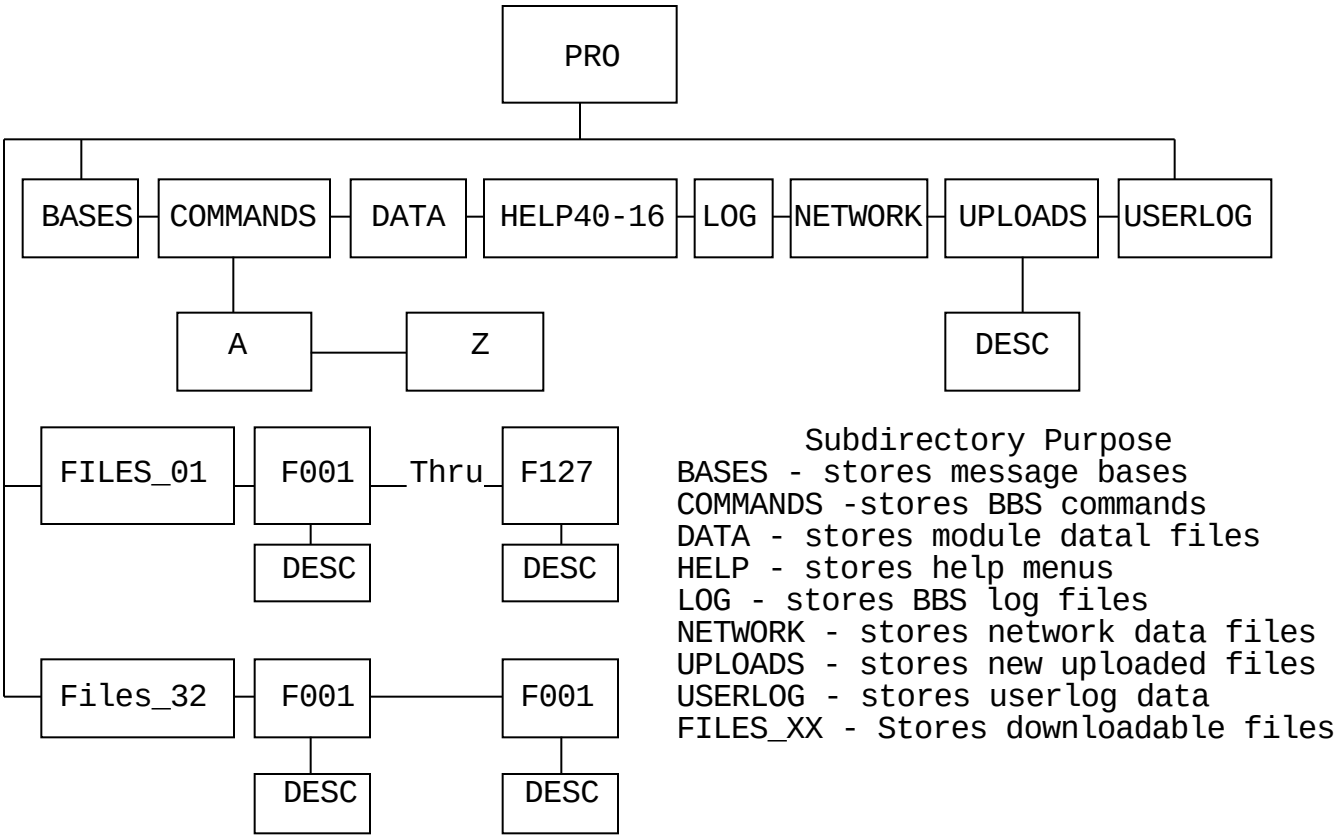
Disk 3 contains the commands that would not fit on disk 2.

Update Package

With the release of BBS Express! Professional V5.0b, all commands that we own are included. You have nothing else to purchase, Bank Mio, Pro_Term, Xmodem, Hi_SpeedNet!, are all included. In addition if you have a MUX, V5.0b is compeletly compatible for Multi-Line use.

The System Directories

BBS Express! PRO uses subdirectories extensively. The entire board and all its files reside under one subdirectory in the main directory. To illustrate this, look at the diagram below. This pictorially represents how PRO's subdirectories are structured.



As you can see, each of these subdirectories reside under the subdirectory called PRO in the MAIN directory. While each of the subdirectories may physically reside on the same drive they don't have to. You can put the COMMANDS directory on drive 8 and the UPLOAD directory on drive 2 and the FILES_XX subdirectories on drives 4, 5 and 6. The key to doing this is that each drive that you will be using with the BBS has a subdirectory called PRO defined in the MAIN directory of that drive and the BBS subdirectories reside in this PRO subdirectory.

To make the task of creating these subdirectories easier, we have provided a batch file on the master disk called **MAKESUB.BAT** which will create all the necessary subdirectories for you. You must edit this batch file with a text editor and change the drive numbers to the drive where you want the various subdirectories located. This batch file will also make the first three subdirectories required for the first three download file areas. You may adjust this to your own requirements prior to running the batch file, depending on how many download file areas you are setting up for your system.

After you have created all the directories, it is recommended that you copy the files contained on the master disks to the appropriate subdirectories you just created. The master disks contain all the files within the subdirectories where they should reside, so a wildcard subdirectory copy is acceptable. Make yourself a list of where you decided to place the subdirectories. This will aid you in the next step of creating the **SYSDATA.DAT** file.

Getting Started:

If you are already running another version of BBS Express! Professional, the first thing you need to do is clean out your **COMMANDS>A** subdirectory. Move these commands to another subdirectory or back them up to disk. Install **ALL** the **COMMANDS** from the **MASTER DISK** into your **COMMANDS>A>**, subdirectory. There are less than 126 commands so they all will fit. In order for your BBS to function properly the new commands are to be called first, if you run a ramdisk then be sure to change the files you are calling there from the newest files you just put into your **A** directory.

If you wish to run **THIRD PARTY** commands, then rename the command in the **A** directory *.INA, this will render it inactive. It is suggested that you only do this one command at a time so that you are able to determine if the third party command will function properly with this version of **BBS EXPRESS! Professional**.

Installation Help Files

Check your master disk for **HELP.BAT**, edit the file to your systems drive specification. Then run the **BAT** file, this will copy your present **HELP40** and **HELP80** into the newly created subdirectories, changing them from ***.ASC/*.ATA** to ***.HLP**. Your HELP40 and HELP80 files are still required due to some programmers hard-coding them into their commands.

Creating the Sysdata.Dat File

The SYSDATA file contains all the system parameters that PRO needs to know where to find things. This file is created and maintained using the SYSEDIT.COM program. SYSEDIT2.COM allows for creating SYSDATA1.DAT rather than renaming the files.

SYSEDIT.COM is a stand alone program which is used to create and later edit the Sysdata0.dat file. This file contains all the system data parameters which Express! Professional uses to determine where required system files are located and how to configure the system setup. Using the program is pretty intuitive, but a detailed explanation of the various options is provided below.

Sysedit.Com is loaded from the SpartaDOS command prompt by typing SYSEDIT and pressing return. Make sure that BASIC is disabled. When the program finishes loading, you will be prompted to enter a drive number for the SYSDATA0.DAT file. If SYSEDIT can not find the sysdata file on that drive, you will be prompted with 'Create A New Sysdata ?'. Respond Yes or No. Responding No will exit back to the DOS prompt and responding Yes will initialize a new sysdata file for editing. When you exit the Syseditor (Option 9 from Main Menu), you will be prompted for a filename to save the configuration, when using a MUXed system you will need a SYSDATA0-SYSDATA7,for each system you run on the MUX, or press return to save under the displayed input filename. The SYSDATA0.DAT file should always be saved in the PRO subdirectory on the drive that you plan to boot the board.

After specifying the drive number for the Sysdata0.dat file, the Main Menu will appear and present you with the following options:

```
          Main Menu
  [1] System Parameters
  [2] Drive Parameters
  [3] Main Commands
  [4] File Sig Parameters
  [5] System Baud Rates
  [6] Misc. Parameters
  [7] ExpressNET Parameters
  [8] Event Scheduler
  [9] Exit Sysdata Editor
```

Each of these options is discussed below. An asterisk will appear to the left of the menu number if you have entered that option during this editing session. While in any option 1-8, you may return to the Main Menu by pressing the Escape Key.

```
[1] System Parameters:
    [A]  System Hi Message          99999
    [B]  Total System Calls         99999
    [C]  Calls Today                99999
    [D]  Feedback Mode              xxxxxx
    [E]  Allow Handles              xxx
    [F]  Interface Type             xxx
    [G]  Use File Sig Desc          xxx
    [H]  Secondary Password
              xxxxxxxxxxxxxxx
    [I]  Board Name for Status Line
    xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
    [J]  Sysop Name for Status Line
    xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

[A] System hi Message

This is the Hi message counter. If you are just starting out with PRO, this field should be set to one. The board increments this number for each message or E-mail posted on the system. Press return after entering a new value.

[B] Total Calls

Total system calls to date that your BBS has received. If just starting a BBS, this number should be left zero. Otherwise, specify your current system calls in this field. Press return after entering a new value.

[C] Calls Today

Total calls your system has received today. Press return after entering a new value.

[D] Feedback Mode

Specifies whether the system will direct sysop feedback to E-mail or is turned off. This field toggles by hitting the option letter.

[E] Allow Handles

This field toggles between YES and NO. It is used to specify whether you want to allow the use of handles on your system.

[F] Interface Type

This field toggles between MIO, BlACK BOX, PRC and 850. Set to the interface type that you are using. If you are using ANY floppy drives on the system, you should not set to the MIO interface type, but rather, you should specify 850 or PRC. This is due to the fact that Atari floppy

drives communicate with the computer through the SIO port. When communicating in this manner to a floppy drive, the modem port must be ‘suspended’.

Only the 850 and PRC settings ‘suspend’ the modem. The MIO setting is intended only for those systems running COMPLETELY from hard/ram disks and utilizing the RS232 port on the MIO.

[G] Use File Sig Desc

This field toggles between YES and NO. Specifies whether you want to use descriptions with download files.

[H] Secondary Password

Specifies the current sysop level password. This field may be up to 15 characters. Any user with sysop access will be required to enter this password when accessing the system remotely. When finished editing this field, press return.

[I] Board Name for Status Line

Specifies a string of up to 36 characters for board name which will display in the top status line while the board is running. This field is automatically centered when editing is terminated with a return.

[J] Sysop Name for Status Line

Specifies a string of up to 36 characters for sysop name which will display in the top status line while the board is running. This field is automatically centered when editing is terminated with a return.

[2] Drive Parameters:

```
[A]  UserLog Drive              Dx:
[B]  Help40 Drive               Dx:
[C]  Help80 Drive               Dx:
[D]  Command Drive              Dx:
[E]  System Log Drive           Dx:
[F]  System Data Drive          Dx:
[G]  Upload Drive               Dx:
[H]  Msg Bases Drives
----+----1----+----2----+----3-E
 00000000000000000000000000000000
[I]  File Sig Drives
----+----1----+----2----+----3--
 00000000000000000000000000000000
```

[A] thru [G]

These options are used to specify the drive number where each of the required system directories may be found. To change drive numbers, press the letter which you wish to change until the drive number you want appears on the screen.

NOTE: Option E allows the drive to be set to 0. If the Log Drive is set to 0, this will turn OFF the logging of callers to the call log.

NOTE: Option D has been replaced by the CMDPATH.DAT file and is no longer used by the BBS.

[H] Message Base Drives

Specifies the drive number where each message base may be found. Enter a drive number (1 - 8) for each active message base. If a message base does not exist or is currently inactive, specify a 0 drive number.

[I] File Sig Drives

Specifies the drive number where each file SIG may be found. Enter a drive number (1 - 8) for each active file SIG. If a file SIG does not exist or is currently inactive, specify a 0 drive number.

[3] Main Commands:

Num	Key	Lvl	Typ	.CMD
#01	x	0	x	xxxxxxxx
#02	x	0	x	xxxxxxxx
#03	x	0	x	xxxxxxxx
#04	x	0	x	xxxxxxxx
#05	x	0	x	xxxxxxxx
#06	x	0	x	xxxxxxxx
#07	x	0	x	xxxxxxxx
#08	x	0	x	xxxxxxxx
#09	x	0	x	xxxxxxxx
#10	x	0	x	xxxxxxxx
#11	x	0	x	xxxxxxxx
#12	x	0	x	xxxxxxxx

The Main Command screen displays up to 35 main command definitions in a scrolling manner. Use the Up/Down arrows to move up and down the commands. When the bottom command is reached and the down arrow is hit again, the next page of commands will appear.

Likewise, when the hi-light bar reaches the top command, the previous page will appear, except if you are at the top of page 1. Position the hi-light bar at the command you wish to edit and press return. You may now change the information which pertains to that command. Each command entry consists of four pieces of information.

Key

Specifies the Key definition. This defines the key as a valid entry from the main command prompt. A dash specifies unused command entries. Any character may be used except dash to specify a valid key. If a dash is keyed, the command is reset to an invalid command.

Lvl

A value between 0 and 32 to specify the security level required to access this key function. Note: 32 is sysop level. This field corresponds to the user's "command Security Level" settings. Example: If you set a command to level 3, the user's command level 3 flag would have to be turned on for them to be able to execute the function.

Typ

Specifies the type of file that is contained in the .CMD field. Hitting return on this field will default to type C.

C - denotes that this is a command (executable .cmd file), or any executable .BAT file placed in the Command directory.

T - denotes that this is a text or menu file to be displayed.

.CMD

Specifies the filename (no extender) which will be viewed or executed if the key is pressed by the user. If only return is pressed on this field, the command edit is aborted and the original command will reappear unchanged. Once the filename is keyed in, press Return if the filename is less than 8 characters long.

[4] File Sig Parameters:

Num	File Sig Name
#01	xxxxxxxxxxxxxxxxxxx
#02	xxxxxxxxxxxxxxxxxxx
#03	xxxxxxxxxxxxxxxxxxx
#04	xxxxxxxxxxxxxxxxxxx
#05	xxxxxxxxxxxxxxxxxxx
#06	xxxxxxxxxxxxxxxxxxx
#07	xxxxxxxxxxxxxxxxxxx
#08	xxxxxxxxxxxxxxxxxxx
#09	xxxxxxxxxxxxxxxxxxx
#10	xxxxxxxxxxxxxxxxxxx
#11	xxxxxxxxxxxxxxxxxxx
#12	xxxxxxxxxxxxxxxxxxx

The File Sig screen displays up to 32 file sig definitions in a scrolling manner. Use the Up/Down arrows to move up and down the file sigs. When the bottom file sig is reached and down arrow is hit again, the next page of file sigs will appear. Likewise, when the hi-light bar reaches the top file sig, the previous page will appear, except when on page 1. To edit any file sig entry, position the hi-light bar at the file sig you wish to edit and press return. You may now change the file sig name. Pressing return without keying any change will leave the original sig name intact and unchanged. After editing the file sig name, press return to terminate editing. Each file sig name may have up to 20 characters. Editing automatically ends if you enter your 20th character, so a return would not be necessary in that case.

[5] System Baud Rates:

[A] MAX Baud Rate..... xxxxx
[B] MIN Baud Rate..... xxxxx
[C] Starting Baud Rate..... xxxxx

[A] Max Baud Rate - Toggles the maximum baud rate. Baud will toggle up to the next baud rate each time the key is pressed. Set the maximum baud rate your modem supports. (300 thru 19200)

[B] Min Baud Rate - Toggles the minimum baud rate. Baud will toggle up to the next baud rate each time the key is pressed. Set the minimum baud rate you wish to support. (300 thru 19200)

[C] Starting Baud Rate - Toggles the starting baud rate. Baud will toggle up to the next baud rate each time the key is pressed. Set to the baud rate you want the phone answered. You can not specify a starting baud rate less than minimum or greater than maximum. Starting Baud rate will automatically be set to maximum baud rate if maximum is 2400 or greater.

This is usually a modem requirement that the port speed be set to the highest baud speed for the modem to answer the phone, so we set this for you if maximum is above 1200. You can not override this feature.

[6] Misc. Parameters:

```
[A] Caller Type...... xxxxxxxxxxxx
[B] Display Columns.. xx
[C] Change Foreground Color
[D] Change Background To Next Color
[E] Change Background To Next Hue
[F] Change Border To Next Color
[G] Change Border To Next Hue
[H] Set Border To Background Color
[I] Reset To Default Color Set
```

[A] Caller Type - Toggles between Atascii Only and Ascii/Atascii. This allows you to set the caller type the board will accept at logon.

[B] Display Columns - Toggles between 40 and 80 columns. (Tech Note: Future enhancement. PRO currently does not use this field.)

[C] Change Foreground Color - Options 'C' thru 'I' function
[D] Change Background To Next Color as indicated. Press the
[E] Change Background To Next Hue appropriate option letter to
[F] Change Border To Next Color perform the desired function.
[G] Change Border To Next Hue The color combination that
[H] Set border To Background Color you set on your screen
[I] Reset To Default Color Set will be the colors displayed while the board is running.

[7] ExpressNET! Parameters:

```
[A] Node Number..... xxxxx
[B] Node Name:
    xxxxxxxxxxxxxxxxxxxxxxxx
[C] Node City/State:
    xxxxxxxxxxxxxxxxxxxxxxxx
[D] Network Drive..... Dx:
```

[A] Node Number - Your BBS node number if you are using the ExpressNET! interface feature. Your node number is the serial number found on your master BBS Express! PRO diskette. This is the number by which other ExpressNET! systems will know you.

[B] Node Name - Your BBS name as you want it to appear in network messages sent to other ExpressNET! systems.

[C] Node City/State - Your BBS City and State as you want it to appear in network messages sent to other ExpressNET! systems.

[D] Network Drive - Specifies the drive number where the NETWORK directory may be found. Pressing this letter will increment the drive number. Setting the drive number to 0 disables the networking feature of the BBS.

[8] Event Scheduler:

Event Type	Schedule Days S-M-T-W-T-F-S	Hr	Event Name
Every	Y Y Y Y Y Y Y	0	WHOMAINT
Timed	N N N N N N Y	22	ULPURGE
Timed	N N N N N N Y	23	ULBACKUP
OFF	N N N N N N N	0	
OFF	N N N N N N N	0	
OFF	N N N N N N N	0	
OFF	N N N N N N N	0	
OFF	N N N N N N N	0	
OFF	N N N N N N N	0	

The Event Scheduler screen allows you to set up events to run automatically either after every call or on a timed basis. Up to 30 events can be set up to run. The arrow keys allow you to move to the entry you wish to edit. This screen, like previous screens, will move to the next page when the down arrow key is pressed and you are on the last entry on that page. The up arrow key will move you to the previous page when you are on the top entry. Once you are positioned to the entry you wish to edit, pressing the return key will place you in edit mode. The line will clear and a prompt will be displayed for the appropriate entry. If option 2 is selected, the cursor will move to the event name. The 8 character name of the event should then be entered (Refer to the Event Scheduler section for a list of the supplied events on the master disk). If the filename is less than 8 characters, press return, otherwise entry terminates on the 8th character entered. The hi-light bar will re-appear to indicate the entry has been saved in memory. If you select the event as a timed event (option 1), you will be prompted for the days and the hour that this event should be run. Hitting a return during days entry will default to a Y entry. The hour should be entered in military time (0-23). If you wish to reset an event to OFF, position to the event entry and hit the return key twice.

The Event Scheduler has been changed so that you may now run any BAT file. Create the .BAT File you wish to run and place it in any of

your command subdirectories.

Creating The Userlog

Create the userlog by running the program MAKEULOG.COM. You will first be prompted to enter the number of users you want to allow in the userlog. This can be any number up to 65535. To aid you in this decision, 1 user record requires 1 double density sector (256 Bytes).

In addition, the system will reserve the first 9 records for it's own use, so the sysop is actually user number 10 and the co-sysop is user number 11. If do not plan to have a co-sysop, the system still reserves user number 11. The board treats these 2 user id's differently from the rest. As an example of this, user record 10 and 11 are not recorded in the "who's called log" for users to see. This way, you and a co-sysop can pop into the board without alerting users that you were there.

Next, you will be asked to enter the drive number on which to create the userlog. This can be any drive 1-8.

The program will now start creating the userlog assuming that the subdirectory PRO>USERLOG> exists on the drive that you specified. If not, the program will exit back to the DOS prompt allowing you to create the subdirectory or rerun the program selecting a different drive number.

Creating The Message Bases

Create the message bases by running the program **MAKEMSGB.COM**. You will first be prompted to enter the message base number that you want to create. Hitting return will create the default message base that is displayed on the screen (this always starts at message base #1). Next, you will be asked for the number of messages that you want in this base. This can be up to 250 messages. Then, you will be asked for the drive number on which to create this base. Any drive 1-8 may be used. Next, you will be asked to name the message base. Up to 20 characters may be keyed. Then, you will be asked if you want to allow ATASCII graphics on this base. Enter a 'Y' or 'N'.

Finally, you will be asked for a size (2-9) of the message base. The following chart will aid you in selecting the appropriate size for your needs.

Size Selected	# Of Messages	# Bytes Per Message	Total Base Size
2	100	362	79,360
3	100	618	104,960
4	100	874	130,560
5	100	1130	156,160
6	100	1386	181,760
7	100	1642	207,360
8	100	1898	232,960
9	100	2154	258,560

The number of bytes per message in the chart above represents the number of bytes available when posting a message. If your base size selected was '5', the user would be able to key up to 1130 bytes into the message.

Assuming a message base of 250 messages and size 9, the maximum size of a message base would be 642,560 bytes.

The program will now start creating the message base assuming the subdirectory PRO>BASES> exists on the drive you specified. If not, the program will exit back to the DOS prompt, allowing you to create the subdirectory or rerun the program, selecting a different drive number. Once the message base has been created, the base counter increments and the process continues for the next message base.

Regardless of how many message bases you decide to have online, remember that you must always create message base 32. This serves as the E-mail base. Once message base 32 is created, the MAKEMSGB program will exit back to the Dos command prompt.

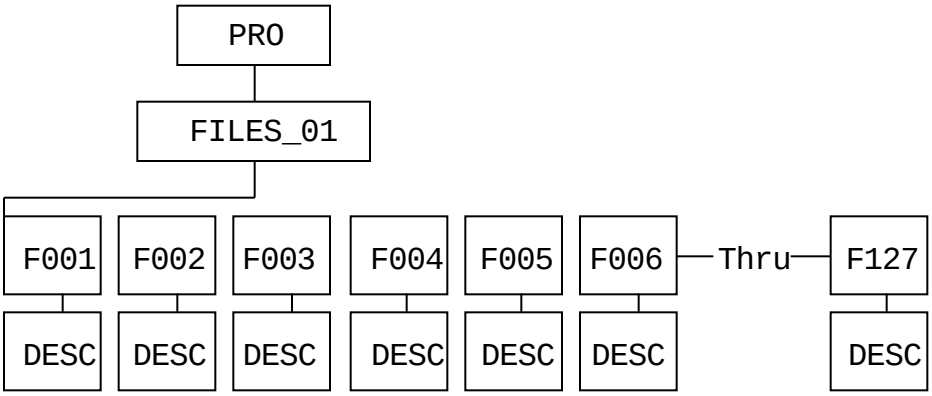
Creating The File Sigs

BBS Express! PRO supports up to 32 download file sig areas and 1 upload file sig. All uploads go to this one file sig area. They are not visible to the user until validated by the sysop. (Refer to the New Files Validation section for the procedure on validating new uploads).

Each download file sig is actually a subdirectory named FILES_XX. XX is the sig number (1 thru 32). Within the FILES_XX subdirectory up to 127 subdirector can exist named F001 through F127. These subdirectories must be created in sequential numeric order. The actual download files are stored in these subdirectories.

Under each of the F001 thru F127 subdirectories are subdirectories called DESC. This is where the description to the file is stored. A quick calculation shows 127 subdirectories, each containing up to 127 files, yields space for 16,129 download files per file sig area. With 32 file sig areas, 516,128 files online at one time is possible.

Let's look at one download file sig area pictorially to illustrate this:



Each of the 32 file sig areas is structured exactly like this example, except the FILES_01 subdirectory would be named FILES_02, FILES_03 etc.

If you initially don't need all the subdirectories F002 though F127, don't create them till you need them. Pro's NEWFILES validation command module will only attempt to use the next subdirectory when the current subdirectory has filled up with 127 files. If you create F002 though F127 and they do not contain files, PRO still has to look in these subdirectories to determine if any files exist. We recommend you stay one subdirectory ahead of what you need. This will afford the best response time to the user.

A word of caution... when creating the F001, F002, F003 subdirectories, don't skip a number. The board will be tricked into thinking there are no more subdirectories in the sig area if it encounters a non-existent numeric subdirectory. You can have FILES_01, FILES_10, FILES_20 with gaps in-between but not within those subdirectories. When creating the additional subdirectories, don't forget to create the DESC subdirectory under it so the file descriptions can be saved. The user can still upload files if you forget, but would see an 'Unable to Create Description File' message when the BBS tries to save the file description.

Creating Help Menus

The necessary help menus PRO requires are on the master diskette in the HELP40 and HELP80 subdirectories. All of these files are text files, and can be edited with any text editor.

PRO has 16 sets of menus available for display. They are based on the caller's video width and translation. The help menus are broken down into 16 subdirectories called HELP40, HELP80, HELP40AT, HELP80AT, HELP40AC, HELP80AC, HELP40VT, HELP80VT, HELP40AN, HELP80AN, HELP40CG, HELP80CG, HELP40V1, HELP80V1, HELP40IG and,HELP80IG. Within each of these subdirectories are 2 sets of menus distinguished by the extenders .ATA for ATASCII menus and .ASC for ASCII menus. (**This is for the HELP40 and HELP80, only!**). **The remaining subdirectories require and extender of .HLP**. When the user calls for a help file to be displayed, PRO will display the appropriate help file based on the user's current settings. This makes it easy to customize the help files for both 40 and 80 column users as well as ATASCII and ASCII users. The help files provided on the disks are generic in nature, but can be used to get you started. They are intended to be a starting point and can be customized to your liking.

HELP.DAT, This files is to be placed in your **D1:>PRO>**, subdirectory along with your **DATAPATH.DAT**, **CMDPATH.DAT**, etc files. This files consists of (20)twenty dashes (-), the twenty dashes represent the **16 Help** subdirectories, 4 are held in reserve for future use. The **SysOp** has the option to use these features or turn them off. To use the subdirectory replace the dash with the drive number of the help directory. The dashes are in order of the above mentioned help directories.

The following table contains the variable tags available for use in menus. To aid in the conversion of existing menus, we have included the 1030/850 variable tag equivalent for those of you running the original version of BBS Express!

PRO Variable Tags Definitions Table

Tag Name	Number	1030/850 Version Equivalent
Clear The Screen	&00	---
Users Handle	&01	&01
Users Real Name	&02	---
Users Password	&03	&02
Users Address	&04	---
Users City/State	&05	&03 &04
Users Zip Code	&06	---
Users Country	&07	---
Users Phone Number	&08	&05
Users Age	&09	---
Users CPU Type	&10	---
Users Lastcall Date	&11	&06
Users Time Limit Per Call	&12	---
Users Time Limit Per Day	&13	&09
Users Minutes On Today	&14	&08
Users Last Read Message	&15	&10
Users DL Ratio	&16	---
Users Downloads	&17	&12
Users Uploads	&18	&11
Users Messages Posted	&19	---
Users E-mails Sent	&20	---
Users Total Calls	&21	&14
Users Video Width	&22	---
System Hi Message	&23	&22
System Total Calls	&24	&24
System Calls Today	&25	&35
System Last Caller	&26	&20
Current Translation	&27	---
Current Date	&28	&19
Current Time	&29	&21
User Number Online	&30	---
Current Baud Rate	&31	&36
Free Time This Call	&32	---
Minutes Connected	&33	&25

Note : Tags numbers 07, 13, 15, 16, 17, 18, 23, 26, 27, 28, 29, 30, 31, 32, 33, 34 have no direct translation to PRO.

These variable tags can be used in your help files in place of hard coded literals. Let's look at an example of a typical welcome help file and see how these variable tags are used.

This is what your help file would look like:

```
&00

        Hi &01...
        Welcome to Inside the 8-bit bbs
        How Are Things In &05?
        Your Last Call Was On &11...
        You Just Missed &26...
```

And this is how the user would see it at logon:

The user's screen would clear due to the &00

```
        Hi John Doe...
        Welcome to Inside the 8-bit BBS!
        How Are Things In Augusta, Ga.?
        Your Last Call Was On Fri 18-Apr-88...
        You Just Missed Stephen J. Carden.
```

All the variable tags were replaced with meaningful information based on their current values. You can use these variable tags in help files, menus or text files anytime you want this variable information displayed to the user. An & with no digit ('0' - '9') next to it will display as a normal '&'. In other words, saying This & That in a menu file will display 'This & That', just as it should.

The master disk contains a complete working set of help files in 40 and 80 column format for both ATASCII and ASCII translation. These should be copied to your PRO>HELP40> and PRO>HELP80> subdirectories and can be used as is, or used as a guideline for creating your own custom files.

Colorizing Your Menus

The BBS shell supports both VT52 and ANSI (IBM) color modes. This color can be used by you in your .ASC menu files (there is no need to place them in your A files). Different .CMD modules also change the user's colors internally.

To use colors in your menus, you simply use an '&' command in the form of '&va' where 'v' is the color to use if the user is in VT52 color mode, and 'a' is the color to use if the user is in ANSI mode. (if the user is not in a "color" mode, then the &va is ignored) .

These colors are signified by a one-character code of the following:

VT52	'b'	Black
	'r'	Red
	'g'	Green
ANSI	'r'	Red
	'g'	Green
	'y'	Yellow
	'b'	Blue
	'm'	Magenta (pink)
	'c'	Cyan (blue-green)
	'w'	White

For example, the following:

```
&gb Welcome to Inside the 8-Bit BBS
```

Would display in 'g'reen if the user is in VT52 mode, and in 'b'lue if the user is in ANSI mode. Once you change a color, that color remains in effect until (a) you change it again with another '&va' command, or (b) the user returns to the MAIN.CMD processor. MAIN.CMD always resets VT52 color to black and ANSI to white letters.

There is also support for doing 'Box' characters on the IBM ANSI terminals. You can use the ATASCII box characters in your .ASC menu files (the ATASCII characters are ^Q, ^W, ^E, ^R, ^A, ^S, ^D, ^Z, ^X, and ^C).

If you use ATASCII box characters and the user is in IBM ANSI mode, then the shell will transform those characters to the corresponding IBM box character before sending them out over the modem. If the user is currently in ASCII or VT52 mode, then the shell transforms the box characters to appropriate ASCII boxes comprised of '+'s and '-'s, for example:

Modifying Your Startup.Bat File

Your Startup.Bat file should include the following SpartaDOS commands in addition to any other commands that you run during your startup.bat procedure.

```
TDLINE.COM
RS232.COM            (not needed if using an MIO)
```

If your ramdisk is of sufficient size, we suggest that you create the subdirectory PRO>COMMANDS>A> (even if you are running with a hard drive) and copy high usage command modules to this subdirectory on the ramdisk. The remaining command modules can be kept on your hard drive. Just be sure to edit the CMDPATH.DAT file and set your ramdisk as the first search drive and your hard drive as your second search drive. The copying of modules to the ramdisk can be performed within the Startup.Bat if you wish.

Considering that the command modules are loaded quite a bit during board operation, placing them in the ramdisk for access will speed up the response time to the caller and reduce the wear and tear on your hard drive by only having to read the commands once from the hard drive each time the system is booted.

Autobooting the BBS from the Startup.Bat file, and thereby making the board self booting can be achieved by making the BBS.COM file the last command in the batch file. Also keep in mind that running a batch file from a batch file terminates the original batch file even though you may have had more commands to execute. You can use this to your advantage by making the board setup batch file a stand alone batch file in your PRO> subdirectory. Adding one line to the end of your Startup.Bat file to run the batch file to configure and run the BBS. Should you need to hit reset to go back to DOS to perform something, you can restart the board by typing BBS.

Configuring The Modem

The exact dip switch settings will vary from modem to modem depending on the manufacturer of your modem. As a guideline in setting up your particular modem to run with PRO, the following criteria should be used in determining the correct DIP settings.

1. The modem should only turn on the carrier detect pin when a remote carrier is present.
2. The modem should hang up and return to command mode when the DTR goes off.
3. The modem should return messages back to the CPU in words.
4. The modem should understand its own commands.
5. The modem should be in Asynchronous mode.
6. The modem should NOT be in auto-answer mode.

Here are some example Dip Switch settings for a few of the popular modems being used today.

Hayes and SmarTeam 300/1200 Baud Modems

Switch 1 - UP
Switch 2 - UP
Switch 3 - DOWN
Switch 4 - UP
Switch 5 - DOWN
Switch 6 - UP
Switch 7 - UP
Switch 8 - DOWN

Avatex 300/1200 (Non HC version)
All Dip Switches should be UP

Avatex 300/1200 (HC version)

Switch 1 - UP
Switch 2 - UP
Switch 3 - DOWN
Switch 4 - UP
Switch 5 - DOWN
Switch 6 - UP
Switch 7 - DOWN
Switch 8 - DOWN

Most of the 2400 baud modems on the market today use internal or `soft set' dip switches. They require that you configure them using a communications program in terminal mode.

With your modem turned on, boot your terminal program and enter terminal mode. Make sure you are set to ASCII translation, Half Duplex and set to the 2400 baud. Key the following commands terminated by a return. The modem should respond with the message 'OK'.

```
AT&F    - resets the modem to factory settings
AT&C1   - turn the carrier detect pin on only when a
          remote carrier  is present
AT&D2   - modem will hang up and return to command mode
          when the DTR pin goes off
ATS0=0  - turn off auto-answer mode
AT&W    - writes the current settings to non-volatile
          modem memory. The modem may be turned off or
          even unplugged and the settings will not be
          destroyed.
```

Once these settings are made, you should never have to reset them.

A special note concerning the Atari SX212 Modem

While PRO will run on the SX212 if you use an interface through the modem RS232 port and load the appropriate handler, the BBS will not run through the SIO port. There are no plans
to convert BBS Express! PRO to run through this modem port.

A special note to Avatex 1200 owners

The Avatex 1200 (non-HC) is a non-standard, yet very popular modem. Unfortunately, it does not understand most "Hayes Compatible" commands. There is a special "WaitCall" module on your PRO disk that was written specifically for the Avatex 1200 modem. This file is named WTCALLAV.CMD. If you plan on running PRO on this modem, you should copy this command module to your PRO>COMMAND>A> subdirectory under the name WAITCAL0.CMD. This special module takes the place of the standard waitcall module.

With V5.0b, the **WAITCALL.CMD**, you use must now be renamed to **WAITCAL0.CMD**. If you are using a **MUX**, system then it must also be copied to **WAITCAL1.CMD**, **WAITCAL2.CMD**, etc.

User Editor and System Security

With BBS Express! PRO, you have COMPLETE control over everything that a user can or can't perform. By now you may have noticed that everything under BBS Express! PRO is broken up into 32's. There are possible maximums of 32 message bases, 32 file SIG areas and 32 command levels.

Additionally, each user's record has groups of 32 on/off flags for different functions. These possible functions are broken up in the following manner.

For the message bases, there are 32 individual flags for:

- Can the user READ messages on this base?
- Can the user POST messages on this base?
- Can the user EDIT *ANY* message on this base?
- Can the user DELETE *ANY* message on this base?
- Can the user PRINT *ANY* message on this base?

For the file area, there are 32 individual flags for:

- Can the user ACCESS this SIG?
- Can the user EDIT DESCRIPTIONS on this SIG?
- Can the user DELETE *ANY* file on this SIG?

For the command security, there are also 32 individual flags to signify whether a user may execute that level command.

As an example, let's assume that we have a user named John Doe that has the following security level flags:

```
 COMMAND:                1111111111222222222333
                123456789012345678901234567890 12
       Commands YY.............................


      MSG BASES           1111111111222222222333
                 123456789012345678901234567890 12

           Read? YYYY..Y.Y.....Y................Y
           Post? YY....Y.......Y................Y
           Edit? YY.............................
         Delete? YY............Y................
          Print? YY............Y................


      FILE SIGS           1111111111222222222333
                 123456789012345678901234567890 12

        Access?  YYYY..........Y.................
        Edt Desc?..............Y.................
        Delete?  .............................
```

Now... What exactly can John do? Well, as far as command levels, he can execute level 1 and 2 commands. Remember, command levels for each individual command are kept in your SYSDATA.DAT file, and can be changed by using the SYSEDIT program. This is probably the right time to explain the importance of NOT giving people 'Sysop Access'. First, always keep the security level of the 'Sysop Command Mode' (in your SYSDATA file) at level 32. Next, NEVER give any of your users level 32 command access unless you are absolutely sure of what you are doing. Why? Well, the reasons are simple. A terribly upset level 32 Sysop could wipe out your ENTIRE hard drive in a matter of minutes. The 'Dos Shell' of BBS Express! PRO is about as powerful as SpartaDOS itself. From within the Dos Shell, it is possible to delete every user in your userlog, delete any file on any of your drives, delete any subdirectory, etc.

As for the message bases on our BBS, John can do the following. He can READ messages on bases 1, 2, 3, 4, 7, 9, 15 and E-mail (#32) (if a user can not read messages on a base, they will be greeted with an 'Invalid Security' if they try to enter that base). John can POST (write) messages on message bases 1, 2, 7, 15 and E-mail. Thus, it's possible (and sometimes very convenient) to allow a user to read messages on a message base without allowing them to write messages or reply to existing messages. This is sometimes the perfect set-up for 'NEW' users.

As for the message bases on our BBS, John can do the following. He can READ messages on bases 1, 2, 3, 4, 7, 9, 15 and E-mail (#32) (if a user can not read messages on a base, they will be greeted with an 'Invalid Security' if they try to enter that base). John can POST (write) messages on message bases 1, 2, 7, 15 and E-mail. Thus, it's possible (and sometimes very convenient) to allow a user to read messages on a message base without allowing them to write messages or reply to existing messages. This is sometimes the perfect set-up for 'NEW' users.

Not long ago, John came to us with a request. You see, John is an ST user, and he asked us if we would add a new message base and file SIG area devoted to ST users. Our reply? Well, as any good Sysop would do, we said 'Sure, John...as long as YOU volunteer to maintain it! You will have to police the message bases for any vulgar messages and edit any file descriptions that you think can be improved'.

Well, John agreed to be Co-Sysop of the ST areas. So, we set up a new message base and new file SIG, both as #15. We then updated John's security levels which are shown above). If you look at John's message base flags under base 15, you will see that he can do just about everything that is possible on a message base. He can read, post, delete, and print messages (in case he wants you to see one). But he can't edit messages. While we like John, we don't want him editing other people's messages and possibly changing something that is said (this is probably unrealistic, but used mostly for demonstration purposes). In the file SIG's. John can only access file areas 1, 2, 3, 4 and 15. Once again, he has much more freedom in the #15 file area. There, he can also edit file description. He does not have the ability to delete any file on that SIG, as we would prefer to do that on our own.

Now that you understand how the security levels control each user's actions, we can discuss two VERY important 'user records' in your userlog file. These records are referred to as the NEW USER RECORD and the VISITOR RECORD. These records are just like any other userlog data record, except that they are solely for your use, and no user can ever log on with them.

Their importance is that they control what happens when a NEW user logs onto your board (either as a permanent user or a visitor). Here is what BBS Express PRO does: when a user logs onto your board as NEW, Express! will ask them if they want a permanent password. If they respond with 'Yes', the NEW USER RECORD contents will be copied into their user record. If they respond 'No', then the VISITOR RECORD contents will be copied into their user record area. So, these records become the base that their records are built upon.

Not every field in the NEW USER and VISITOR records are important. For example, any data that the user is prompted for (such as handle, name, address, etc) will overwrite what was in the new user/visitor record. The most important fields in the new/visitor records are the security levels and the time limits. Here is a list of the fields that you SHOULD update in your new user and visitor records:

Time limit per call
Time limit per day
Download ratio

Command Security Level
Message Base Security Level
File SIG Security Level

Now let's look at the various functions in the user editor. The user editor can be accessed from either the main menu (if you have it defined in your main command table) or from the Dos Shell by typing UEDITOR from the Dos Shell's command prompt or from the "Waiting For Call" screen.

This is what you see when the UEDITOR first loads:

```
                Edit Which User?

    [U] by User number      [O] Online user
    [H] by Handle           [F] First new
    [R] by Real name        [A] Add new user
    [V] Visitor rec         [1-7] Edit mask
    [N] New User rec        [X] Exit

                Your choice:
```

If you edit a user by one of the above options, that user record is read into memory and the following menu is displayed to you:

```
Editing: SYSOP (#10)
 Status: Active/Validated/LOCKED

        Edit What?

[T] Textual data    [L] Lock user
[U] Usage data      [D] Delete user
[C] Command sec.    [1-7] Apply mask
[M] Message sec.    [X] Exit
[F] File sec.

       Your choice:
```

If you select option T to edit textual data you will be presented with the following:

```
Editing: SYSOP (#10)
 Status: Active/Validated/LOCKED

 [A]    Handle: SYSOP
 [B]    Real Name: Keith Ledbetter
 [C]    Address: 1234 Any Street
 [D]    City: Midlothian, VA
 [E]    Zip Code: 23113
 [F]    Country: USA
 [G]    Phone: 379-4156
 [H]    Computer: Atari (all)
 [I]    Age: 30
 [J]    Password: atari

 A-J, OK, or LIST:
```

If you select option U to edit usage data you will be presented with the following:

```
          Editing: SYSOP (#10)
           Status: Active/Validated/LOCKED

           [A] Time/call : 255
           [B] Time/day  : 255
           [C] Mins today: 1
           [D] Downloads : 0
           [E] Uploads   : 1   (give them 1)
           [F] DL ratio  : 255:1
           [G] Last read : 1
           [H] Msg posted: 0
           [I] E-mail sent:0
           [J] # of calls: 1
           [K] Last call : Mon  18-Apr-88

            A-K, OK, or LIST:
```

If you select option C to edit command levels you will be presented with the following:

```
           Editing: SYSOP (#10)
           Status: Active/Validated/LOCKED

           Which command levels can this
           user execute?

              (Y, N, or Return)

                    1    1    2    2    3
               ----5----0----5----0----5----0--
       Now: yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
       New:
```

If you select option M to edit message base security data you will be presented with the following:

```
   Editing: SYSOP (#10)
    Status: Active/Validated/LOCKED

   What message base actions can
   this user perform?

                1    1    2    2    3
     ----5----0----5----0----5----0--
A] Rd yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
B] Wr yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
C] Ed yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
D] De yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
E] Pr yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy

A-E, OK, or LIST:
```

If you select option F to edit file SIG security data you will be presented with the following:

```
Editing: SYSOP (#10)
Status: Active/Validated/LOCKED

      What file SIG actions can
      this user perform?

                1    1    2    2    3
     ----5----0----5----0----5----0--
A] Rd yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
B] Ed yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy
C] De yyyyyyyyyyyyyyyyyyyyyyyyyyyyyyy

A-C, OK, or LIST:
```

Message Bases

The message bases are accessible through either the = command to go to any individual message base or through the Q command to perform a Quickscan of any messages on all the bases since the user's last call.

The command prompt in the message base processor contains 4 pieces of information. Here are a few sample message base prompts.

(1\50+) General:
(50\50-) General:
(Q\1\50) General:

The first parameter specifies either the read direction that you are currently in ('+' forward or '-' backward) or 'Q' to signify that you are in QuickScan mode. The next parameter specifies the current message number within the message base that you are on. This is the message number that any of the base commands will act upon. The <Return> key always takes you to the next message number. The next parameter specifies the number of messages currently active in this message base. The last parameter specifies the name of the message base you are currently in.

Access to a message base is determined by the user's message base security flags. If the READ flag is set for a base, they can read messages on that base. if the POST flag is set, they can also post messages on that base. A user can always edit or delete a message that they have posted, but would require the appropriate security flag be turned on to perform those functions on a message written by someone else.

Printing a message to the local printer requires that the PRINT security flag for that message base be set ON. This option is available for use by using H key in the message base processor. This function does not appear on the menu. The print option has been modified giving the SysOp a choice of printing hard copy or to disk. If printed to disk it will append to the file created and rest in the LOG subdirectory until deleted by the SysOp.

Should the need arise from time to time, the SysOp may change the name of the message base by using the Z key. You will then be prompted for a new message base name. This option is only functional if the command level 32 security flag is turned on. The SysOp may also move message from one base to another base of the same size with a _, the SysOp can also access the DOSSHELL and the USEREDITOR from the message base area. They may also track networked messages using a Y function key within the message base area. Pressing a **M** will display the size and other pertained information.

Download File Sigs

Each of the possible 32 download file sig areas is accessible to the user if their file sig security bit has been turned on by the sysop. If a user can get to the file area, then they will be able to see ALL files in that area. The security is at the FILE AREA level. You either can or can't get to the area. If you can, you can see everything.

Listed below are the 14 functions that can be performed while at the file area command prompt.

```
[B] Browse files with descriptions
[Q] Verbose listing of files
[C] Catalog files, 10 per page
[N] New files since your last call
[R] Raw (SpartaDOS) catalog listing
[S] Browse files since specified date
[U] Upload a new file
[K] Clear marked files
[V] View marked files
[Y] Ymodem Batch marked files

[/] Go to the next file area
[-] Go to Previous File area
[=] Go to another file area
[X] Exit to the main menu
```

Option [B] Browse files with descriptions

If the user selects option B, they will be presented with one file at a time for viewing. Like this:

```
File: FILENAME.COM
Size: 255 bytes (2 sectors)
Date: 1-31-88  2:09p
Time: 1 min at 9600 baud
Owner: Uploader's Name
D/L's: 45

this is the description for filename.com

[D]ownload  [R]ead       [E]dit Desc
[A]gain     [C]ontinue   [Q]uit
[T]rashcan  [U]NARC      [M]ark
```

The user may select any of these options, except E and T. These two options will only function if the user's security bit for editing descriptions and deleting files for this file sig has been turned on.

The mark option flags the file for downloading with the Ymodem batch protocol, which is accessed using the Y option from the file sig command prompt. Any number of files may be marked for batch downloading. Batch downloading observes the users time limits and download ratio and will bypass files which take longer than the user has time available or if the user has exceeded their download ratio.

If the user selects the download option, they will be prompted with:

```
[X] Standard/Crc xmodem
[W] Windowed Xmodem
[Y] Y-modem (xmodem-1k)
[S] SEAlink
[L] L-modem
[Q] Abort transfer
```

Once the user makes a download protocol selection, the file transfer begins.

Option [C] Catalog files, 15 per page, (27 if in 80 col)

If the user selects option C, they will be presented with up to 15 files at a time for viewing. Like this:

```
    Filename     Size   Date
[A] FILE1    COM   5200  1-31-87
[B] FILE2    COM   5300  1-31-87
[C] FILE3    COM   5400  1-31-87
[D] FILE4    COM   5500  1-31-87

D)ownload R)ead M)ark Q)uit C)ont:
```

If the user selects the download option, they will be prompted with:

Download Which File (A-D)?

If the user selects the mark option, they will be prompted with:

Mark Which File (A-D)?

The user makes their file selection for downloading by letter and will be prompted with:

```
[X] Standard/Crc xmodem
[W] Windowed Xmodem
[Y] Y-modem (xmodem-1k)
[S] SEAlink
[L] L-modem
[Q] Abort transfer
```

Once the user makes a download protocol selection, the file transfer begins.

New Files Validation

When a user uploads a file to your system, it is always sent to the PRO>UPLOAD> subdirectory. The file remains there, inaccessible to the user, until you date the file. If the user entered a file description, it is stored in the PRO>UPLOAD>DESC> subdirectory.

To validate new uploads, go into the Dos Shell and execute the command module NEWFILES. NEWFILES will scan the upload directory and present you with each uploaded file one at a time like this:

```
 File: FILENAME.COM (255 bytes)
Owner: Uploader's Name
  SIG: The SIG name
 Date: 1-31-88  2:09p
 DL's: 0

 this is the description for filename.com

 [V]alidate  [R]ead File  [E]dit Desc
 [D]elete    [N]ext File  [C]hange Name
 [Q]uit

 [Function]=>
```

Validate Option will allow you to validate a file. You will then be prompted with: Move to which file area (0=List)?

You may enter a 0 to see a list of your download file areas or specify a file area to move the uploaded file. The NEWFILES module will scan the file area specified, looking for an available subdirectory to move the file. It will start looking in the F001 subdirectory. If there aren't 127 files in that subdirectory, the file will be moved there. If full, NEWFILES will continue looking thru F002, F003 etc. until it finds a subdirectory with room to move the file. Once an available subdirectory is located, the file and it's description are copied into the subdirectory. The original file and description is deleted from the upload subdirectory. The next upload file is displayed and the process repeats itself until all the files in the upload directory have been presented. Once all uploads have been displayed, you are returned to the Dos Shell command prompt.

Read File
Option will display the file to the screen (and modem if logged on remotely) and then redisplay the file entry.

Edit Desc
Option will allow you to edit the current file description or create a new one if none exists.

Delete Option
will allow you to delete the file without validating it. Both the filename and file description will be deleted.

Next File Option
will allow you to continue to the next upload file without disturbing the current file presented.

Change Name Option
will allow you to rename the file before validating it. Both the filename and file description are renamed.

Quit Option
will allow you to abort new files validation prior to viewing all new uploads.

Browse General Info

These are the Command file's that are needed for Browse.cmd to work:

Browse.cmd	Lmsend.cmd	SEAsend.cmd
Xmsend.cmd	Wxsend.cmd	Ymsend.cmd
Xmbatch.cmd	Arcv.cmd	Zoov.cmd
Zipv.cmd	Lzhv.cmd	Get_path.cmd
Xmodem.cmd	Editdesc.cmd	Allrecv.cmd
Sfxfer.cmd	Ybsend.cmd	

XMODEM.CMD is a command module available to the sysop for use directly in the Dos Shell or in Command Menus. The command syntax is:

Xmodem [S or R] Filename

S - tells PRO to SEND the file
R - tells PRO to RECEIVE the file
M - tells PRO to setup a batch download

Filename - is the FULL pathname of the file.

YBSEND.CMD is a command module available to the sysop for use directly in the Dos Shell. This command allows you to perform Ymodem batch transfers directly from the DosShell.
The command syntax is:

Ybsend Filename (wildcards allowed)

GET_PATH.CMD is called by xmodem.cmd, but it can also be called by the Dosshell or a command_menu.

Syntax for use:

get_path dn:>path>mask dn:>path>filename.ext

The first parameter, "dn:>path>mask", contains the mask of the directory you wish to scan.

The second parameter contains the output name of the batch file you wish to create.

If the user has a level 32 "SYSOP level" then it will prompt him/her if they wish to delete a batch file. It will also prompt them if they wish to get another directory.

SFXFER.CMD: Syntax for use:

Sfxfer s dn:>path>filename.ext (called internally by xmodem.cmd)

ALLRECV.CMD: (Called internally by xmodem.cmd)

EDITDESC.CMD Only minor changes have been made to this file. The clear screen has been removed so the batch description could be employed.

Dosshell usage:

editdesc dn:>path>desc>filename.ext

SPECIAL NOTE:

the Express Cart Version 1.2 Has a bug in when uploading SEAlink, L-modem, and L-modem batch. Keith Ledbetter has been informed of the bug in the cart. The Express Cart v1.1 work's fine when upload any of the protocol(s)

The Dos Shell

The Dos Shell is an environment where board maintenance and most of the major SpartaDos functions are performed. The board comes configured to you with the function accessible from the command prompt by simply pressing the $. From the "Waiting For Call" screen press option 8. There are currently over 50 commands available in the Dos Shell. A help menu is available while in the Dos Shell. Press ?. As new commands became available, it's a simple matter to copy the command into the PRO>COMMAND>A> subdirectory. The command is automatically available to you from the Dos Shell. You can add commands (.CMD files) to the command subdirectory (A thru Z) up to the subdirectory limit. The best way to get a feel for what the Dos Shell can do is to use it once you have the board running.

The commands function as they would from SpartaDos, with a few new ones, like COUNT and SHUTDOWN, It is also capable of running any .BAT file. COUNT simply counts the files and subdirectories in there current path and displays the statistics back to you. (Handy for knowing if a subdirectory is getting close to filling up.) SHUTDOWN, if executed, will effectively lock the BBS program and will keep it from responding to incoming calls. It does not turn the power off or return the system to the DOS prompt, but rather disables the BBS from processing incoming calls until a key is hit from the local keyboard. Once you have shutdown the system remotely, you cannot re-enable it remotely. That can only be done from the local keyboard.

A safety feature of the Dos Shell is if you enter it locally while a user is online and allow the user to see what you are doing, they will not be able to execute any commands because the Shell will not accept a return from the modem. They would be able to key in commands but when they hit return, the Shell would tell them that commands can only be executed locally. It allows them to key in a command while you locally execute a return for them. You can temporarily set them up with command level 32 to allow them to perform functions in the Shell. Don't forget to turn their command level 32 off after the call or they could enter the Shell on their next call and perform functions unattended. This feature was designed to stop a user from attempting to send a DELETE *.* macro across the modem before you could react to it. This only occurs if the Dos Shell was invoked locally. If you or a co-sysop call in and invoke the Dos Shell remotely, all functions are enabled remotely as if you were at the local keyboard.

The Dos Shell's "command line parser" is extremely intelligent. For example, here are various ways you could enter the same command:

```
COPY D1:>PRO>COMMANDS>*.* D2:>PRO>COMMANDS>*.*
COPY D1:>PRO>COM*>*.* D2:>PRO>COMMANDS>
COPY D1:>PRO>COMMANDS> D2:>PRO>COM*>
```

The Dos Shell command is also MS-DOS compatible and accepts commands as CD, \, MD, etc and converts them to Atari format.

```
    BBS Express! Professional Dos Shell Menu
                K-Products (c) 1991
[ ] Denotes Optional Parameter Directory Commands

Dx:              - Change Drive
Dir   [Dx:]      - Long Dir.
Dirs  [Dx:]      - Short Dir.
Pdir  [Dx:]      - Print Long Dir
Pdirs [Dx:]      - Print Short Dir
Credir Path      - Create Dir.
Deldir Path      - Delete Dir.
CWD    Path      - Change Dir.
Count [Path]     - Count Files
        File Commands
Erase  Dx:Fn     - Delete File
Delete Dx:Fn     - Delete File
Rename Dx:Fn     - Rename File
Type   Dx:Fn     - View File
View   Dx:Fn     - View File
Copy   Src Dest  - Copy File
Copy/N Src Dest  - Copy File
 (auto overwrite existing file)
Print  Dx:Fn     - Print File
Lock   Dx:Fn     - Lock Disk
UnLock Dx:Fn     - Unlock Disk
         Disk Commands
ChkDsk   [Dx:]   - Disk Statistics
Protect   Dx:    - Protect File(s)
Unprotect Dx:    - Unprotect File(s)
         Maint. Commands
Ueditor          - User Editor
Editfile [Dx:Fn]- File Editor
Newfiles         - Validate Uploads
Viewevnt         - View Event Status
ULPrint          - Print Ulog Utility
ULBackup         - BackUp Userlog
         Time Commands
Settime          - Set Time
Setdate          - Set Date
Showtime         - Display Time
Showdate         - Display Date
          Misc.Commands
Chat [On/Off]    - Turn Chat On/Off Or Display
                     Current Setting
ShutDown         - Deactivate BBS
Help or ?        - Help Menu
Exit or X        - Exit Sysop Mode
```

NOTE: From the DOS shell, any external command may be run by simply keying in its name (ie: FILESTAT or UEDITOR).

Event Scheduler

The Event Scheduler allows programs requiring no user input to process automatically at predetermined time intervals. The events themselves, as well as time intervals, are set up using Option 8 in the SysEditor. PRO allows you to configure up to 30 of these events to run, based on the type of event you set it up as. As an example, the system automatically configured the WHOMAINT event to run after every call. WHOMAINT is the module that records the caller to the last 50 callers log. On the master disk are 3 event modules excluding the NETWORK modules.

They are :

ULBACKUP.CMD - backs up your userlog

ULPURGE.CMD - deletes users based on lastcall date

WHOMAINT.CMD - records user to the last 50 callers log when they log off the system. This module was automatically set up for you when you initialized a new Sysdata.Dat file.

Each of these modules performs a function that does not require any input from the sysop to carry out its function. While these commands or .BAT files, could be execute from the Dos Shell by the SysOp, the event scheduler removes the burden of remembering to perform these functions. Set them up and forget about them. Let the event scheduler keep track of when these modules should be run.

Normally, after each caller hangs up, as part of the reset procedure, PRO will start the event scheduler. The event scheduler will scan the event list, perform each function if it's time, and finally return control back to PRO. If the board has been idle, Pro will automatically start the event scheduler at the top of each hour, so events will not be missed due to this idle time period. This will occur frequently in the early morning hours when callers are at a low point. This is ideally when you would want to schedule backups, userlog purge or packet processing, if you are using the ExpressNET! networking facility.

The scheduling of events is normally a `set and forget' operation. This is the reason we included the event scheduler in the Syseditor. There is currently no provision for modifying the event schedule while the board is running. To do this, you have to reset the board and run the SysEditor, then rerun the board. We will be writing an online scheduler so this will not be necessary. This module will be placed on the support boards for you to download.

Library Text_Menus

BBS Express! PRO incorporates a text menu structure which permits the sysop to build selection menus for viewing text files, executing commands or even building a text based adventure game. This menuing system is called internal menuing. There is another menu system which is called external menuing which we will discussion in the MENU.CMD section of the manual. There are advantages to both types of menuing systems and it is left up to the sysop to decide which one they prefer to use.

The structure of these internal text menus is broken down into 5 sections.

They are:

1. Command Line
2. Prompt Line
3. User Option Selections
4. END Statement
5. Text Display

Here's a sample Text_Menu file:

```
Text_Menu Level 20 NOABORT Return
View Which Help File? ==>
MAIN D1:PRO>TEXT>MAINHELP.TXT
BASE D1:PRO>TEXT>BASEHELP.TXT
EDIT D1:PRO>TEXT>EDITHELP.TXT
C/R  Exit
END
&00
                 Board Help
                ----------
   [Main]   Help On The Main Commands
   [Base]   Help On The Message Base Commands
   [Edit]   Help On The Message Editor Commands
<Return> To Exit
```

Here's a sample Command_Menu file:

```
Command_Menu Level 20
Play Which Game? ==>
Game1 Cards
Game2 Dungeon
Game3 Hangman
Exit  Exit
C/R   Score
END
&00
```

```
          Online Games

   [Game1]  Play Cards
   [Game2]  Play Dungeon
   [Game3]  Play Hangman
   [Exit]   Return To Command Prompt
   <Return> To See The ScoreBoard
```

Let's discuss each section in detail.

[1] The Command Line - this line tells PRO how to handle this particular text menu. If more than one parameter is used on this line, they should each be separated by a space. The parameters available for use on this line are:

COMMAND_MENU - this option lets PRO know that the selection options are executable (.CMD) files. PRO will attempt to load and execute the file. Once the command executes, the user is returned to the command prompt.

TEXT_MENU - this option lets PRO know that the selection options are text files. PRO will display the selected file to the user and on the local display screen.

RETURN - this option tells PRO to return to this menu when finished displaying a file. Return only works with the text_menu parameter and is only one level deep. So if you set up a text menu with a return option, and it executes another menu with a return in it, the user would return to the second menu. The return to the first menu was broken by the second menu. Using the same first menu, if a second menu is invoked that does not have a return on its command line, then the return would be to the first menu.

LEVEL nn - this option tells PRO to only execute this menu if the user's security level is set ON. 'nn' is the command level required to execute this menu. The user's security does not permit them to execute the menu, they would receive an 'Insufficient Authority' message and would exit back to the command prompt.

QUIET nn - this option functions like the Level nn except that if the user does not have the security level required to execute the menu, they would be returned to the command prompt without receiving any message. This is useful if, let's say, you wanted to alert club members to the next meeting date, but didn't want non-club members to know the menu was attempted.

NOABORT - this option tells PRO to ignore any abort key the user may hit and continue displaying the file. This is useful if you want the user to read you system news EVERY time or have paid advertisements that you don't want the user to be able to abort.

[2] The Prompt Line - this line allows you to set up the prompt that is displayed to the user after the menu display text is presented to the user. This allows greater flexibility for requesting user input depending on the purpose of your menu. Some example prompts would be:

```
Enter Your Selection ==>

[N]orth [S]outh [E]ast [W]est  Which Way? =>
```

[3] The User Option Selection - this section of the menu consists of the actual filenames for each of the selections available in the menu. Each selection in the menu would have a line in this section to

1. Define the selection mask AND
2. the filename associated with this selection

For instance, let's use our sample menu which is a menu containing user help text files for the board. In this section of the menu, we would have 4 entries:

```
MAIN D1:PRO>TEXT>MAINHELP.
BASE D1:PRO>TEXT>BASEHELP.TXT
EDIT D1:PRO>TEXT>EDITHELP.TXT
C/R  Exit
```

Looking at the first line, if the user keys the word MAIN, PRO will display the text file D1:PRO>TEXT>MAINHELP.TXT. If the user keys EDIT, PRO will display the text file D1:PRO>TEXT>EDITHELP.TXT. The C/R is a special option which allows you to define the action which will be taken if the user hits the return key. In this case, we have defined the return to be EXIT, but it can be a text file or even another menu. Some type of exit must always be defined or the user would never be able to get out the menu. In the sample menus above, we have shown you 2 ways to code in an exit. Either using the C/R as an exit or explicitly coding the word EXIT as an option.

When building a Text_Menu, the FULL pathname must be specified including the drive, subdirectory and filename. If you are building a Command_Menu, only the 8 character filename is specified. PRO will always build the full pathname by adding the .CMD to the filename you specified and will always go to the PRO>COMMANDS> (A thru Z) subdirectory to find the file to execute.

You may also pass parameters to a command from a Command_Menu (if the command can take parameters). For instance:

```
Command_Menu Level 32
Download Which File? ==>
Call Xmodem S D1:PRO>LOG>Call.Log
Ulog Xmodem S D1:PRO>Userlog>Userlog.Rpt
Exit  Exit
END
&00
```

```
          Download Board Reports
            Co-Sysop's Only

         [Call]    Download Call Log
         [Ulog]  Download Userlog Report
         [Exit]    Exit This Menu
```

In the above example, the Xmodem command requires 2 parameters be passed to it. The first parameter 'S' tells xmodem to send a file and the second parameter is the filename to send.

[4] End - this parameter terminates the structured menu information from the text display which follows. Everything after this statement to the end of the menu file is displayed to the user. This END parameter MUST be present in the menu or the menu will not work correctly.

[5] Text Display - this is the portion of the menu which is

displayed to the user. Variable tags may be used in this section of the menu. In our sample, used the &00 variable tag to clear the screen before displaying the menu to the user.

CMDPATH.DAT File

The CMDPATH.DAT file is a text file which resides in the same subdirectory as the SYSDATA.DAT file. BBS.COM reads this file at bootup time to set up the order in which the drives are searched for the COMMANDS subdirectories. When booting V5.0b for the first time and this file is not found, PRO will prompt you in creating the file.

The layout of the CMDPATH.DAT file is a one-line entry that is 8 characters long (it must be 8 characters!). The line is nothing but a series of numbers specifying the order in which the drives are to be searched, with a `-' meaning "ignore this entry".

Let's assume that you want to run the most-used commands out of your RAMdisk which is D8:, and the rest from your hard drive which is D1:. This is what your CMDPATH.DAT would look like:

81------<return>

NOTE: It is acceptable to change this file using EDITFILE while the board is running, but you will have to hit RESET and then re-run the BBS for the changes to take affect, as this file is only read in at boot-up.

DLTIME.DAT File

The DLTIME.DAT file is a text file which resides in the same subdirectory as the SYSDATA.DAT file. BBS.COM reads this file at bootup time only.

The DLTIME.DAT file is simply a single line with a number from 0-255 specified. This number represents the amount of tolerance that you want to give users are attempting to download a file that would take them over their time limit (either their "time left this call" or "time left today" limit).

So for example, if your DLTIME.DAT looked like this:

15 <return>

You are saying "allow users to download files that would take them up to 15 minutes over their remaining time". If a file would take them 16 minutes over their time limit, the user would be told "You don't have time to do that!".

NOTE: It is acceptable to change this file using EDITFILE while the board is running, but you will have to hit RESET and then re-run the

BBS for the changes to take affect, as this file is only read in at boot-up.

DATAPATH.DAT FILE

The DATAPATH.DAT file is a text file which resides in the same subdirectory as the SYSDATA.DAT file. BBS.COM reads this file at bootup time only. The DATAPATH.DAT file is never actually used by the BBS, but is intended to be used by user written modules when searching for the PRO>DATA> subdirectory. If this file is not found V5.0b will prompt you to create the file.

The layout of the DATAPATH.DAT file is a one-line entry that is 8 characters long (it must be 8 characters!). The line is nothing but a series of numbers Specifying the order in which the drives are to be searched, with a '-' meaning "ignore this entry".

This is what your DATAPATH.DAT would look like:

814-----<return>

In the above example, you are saying to search drive 8 for the DATA subdirectory, then drive 1 and then drive 4 last. This allows programmers writing modules for PRO to find where you have placed your data subdirectory. The DATAPATH.DAT is read into memory and is accessible to PRO programmers. Refer to the programmers doc's on the master disk for more information on this subject if you are programming modules to run under PRO.

NOTE: It is acceptable to change this file using EDITFILE while the board is running, but you will have to hit RESET and then re-run the BBS for the changes to take affect , as this file is only read in at boot-up.

OVLPATH.DAT File

The OVLPATH.DAT file is a text file which resides in the same subdirectory as the SYSDATA.DAT file. BBS.COM reads this file at bootup time only.

The layout of the OVLPATH.DAT file is a one-line entry that is 10 characters long (it must be 10 characters!). The line is nothing but a series of number specifying the order in which the drives are to be used when saving overlaid memory.

This is what your OVLPATH.DAT would look like:

88844411111<return>

In the above example, you are telling the BBS "put the first 3 overlays on D8:, the next two overlays on D4: and the remaining on D1:. You must always specify 10 overlay drives, even though you will RARELY use over 2 overlays. For example, to use 5 overlays at once, the following would have to happen:

```
program A overlays program B          (creates OVERLAY.0)
program B overlays program C          (creates OVERLAY.1)
program C overlays program D          (creates OVERLAY.2)
program D overlays program E          (creates OVERLAY.3)
program E overlays program F          (creates OVERLAY.4)
```

Each OVERLAY.x file is 16K in size, and we would highly recommend that you put at least the first one on a RAM disk, since BROWSE uses overlays to call EDITDESC, XMODEM, XMBATCH, etc. At most, allocate the first two overlays to RAMdisk, since you'll rarely have a program that goes over two overlays deep. Make sure that the drives you select for overlays have enough free space to hold the OVERLAY.x files.

If this all is confusing you, don't fret. Overlays are hard for non programmers to understand, and luckily you don't need to understand them. Just set up OVLPATH.DAT to "8111111111" and forget it (assuming that D8: is your RAMdisk).

NOTE: It is acceptable to change this file using EDITFILE while the board is running, but you will have to hit RESET and then re-run the BBS for the change to take affect, as this file is only read in at boot-up.

HELP.DAT File

With Editfile command edit the twenty dashes designating the drives your HELP subdirectories are on ie: 3333333---3333333--- this shows the first 7 HELP40 subdirectories with 3 dashes as reserves the other seven dashes represent the HELP80 subdirectories turned on, with three dashes held in reserve for future use. This file goes on the same drive under >PRO> subdirectory your SYSDATA.DAT file is located on. Each time you make a change to this file you will have to re-boot the system for the data file to take effect.

Main Menu Commands

Below are the commands available to the user at the main command prompt. These are the commands set up by default. You may add or delete commands as you. Refer to the section on the Syseditor for the proper procedure for adding main commands.

```
      BBS Express! Professional
               Main Menu
  (Upload Your Files Under [B]rowse!)
*************************************
* A ASCII/ATASCII | B Browse Files   *
* C Call Sysop    | D D/L Locator    *
* F Feedback Area | G Goodbye (Off)  *
* I System InfoNET| L Library Files  *
* N System News   | O Ordering Form  *
* P Parameter Edit| R Read *E-Mail*  *
* S Send *E-Mail* | U User Search     *
* W Last 50 Calls | Y Your Own Stats *
* = GoTo Any Base | * Msg Base List  *
* Q Quickscan!    | @ Edit scan list *
*                 | ! SIG Statistics *
*************************************
*    & - On-Line Upload Validator    *
*    % - On-Line User Editor         *
*    $ - SysOp-Only DOS Shell        *
*************************************
```

Message Base Commands

The following is a list of commands available to the user when they are in a message base.

<Return> reads the next message

[A] Read the current message again	[Q] Quit reading current thread
[B] Set read direction to backward	[R] Reply to the current message
[C] Continue keyword search	[S] See replies to current message
[D] Delete the current message	[T] See thread of current message
[E] Edit the current message	[U] Unlock the current message
[F] Set read direction to forward	[V] Verbose mode toggle (on/off)
[G] Goto a certain message number	[X] Exit back to the main menu
[J] Goto a certain message number	[*] List of active message bases
[K] Search for a keyword	[@] Add/remove base from scan list
[L] Lock the current message	[+] Read ahead one message
[M] Message base statistics	[-] Read back one message
[N] Position at first new message	[>] Read ahead one message
[P] Post a new message	[<] Read back one message

If verbose is OFF, hitting 'A' will read the message

Two additional commands are available to the sysop:

[H] HardCopy Print	[Z] Change name of message base
[_] Move Message to Base ?	[Y] Track Networked Message
[$] Enter Dosshell	[U] Enter Usereditor

Message Editor Commands

The following is a list of commands available to the user in the message editor. Each of these commands is preceded by a '/' in the first position of a.

[A] Abort this message and exit	[L] List the message
[B] Begin this message over again	[N] List the message with numbers
[C] Display a columnar heading	[R] Repeat the last input line
[E] Edit an individual line	[S] Save this message and exit
[F] Formatted listing of message	[U] Unformatted Save and exit
[G] Toggle graphics mode on / off	[W] Toggle word wrap on and off
[I] Insert a new line	[/] Clear the screen

Valid Control Keys

^L - Paste Letter from saved line	^W - Paste Word from saved line
^N - Inserts your Name (handle)	^R - Inserts your Real name

File Sig Commands

The following is a list of commands available to the user when they are in a file sig area.

[B] Browse files with descriptions
[C] Catalog files, 10 per page
[N] New files since your last call
[R] Raw (SpartaDOS) catalog listing
[S] Browse files since specified date
[U] Upload a new file

[K] Clear marked files
[V] View marked files
[Y] Ymodem Batch marked files

[/] Go to the next file area
[=] Go to another file area
[X] Exit to the main menu

Adding System Commands

System commands can be added at any time by simply copying the .CMD file into a command subdirectory. By doing this, they are immediately available to the sysop from the Dos Shell. If it is a command file that will be used by your board users, you will need to define the command in the sysdata file by using the syseditor and editing the main commands. Refer to the section on the Syseditor for the proper procedure for adding main commands.

Starting The Board

Ok...now that we have everything configured, we're ready to run the board. But first, a checklist.

1. Subdirectories created
2. Sysdata.Dat configured (must be in path you boot the BBS from)
3. Userlog created
4. Message Bases created
5. File Sig directories created
6. Help files created and/or edited
7. Help.Dat file edited for your drive configuration
8. Modem configured
9. CmdPath.Dat edited for your drive configuration
10. DLTime.Dat edited with download time allowance value
11. DataPath.Dat edited for your drive configuration
12. OvlData.Dat edited for your drive configuration
13. TDLINE installed (manually or from the Startup.Bat)
14. RS232 installed (manually or from the Startup.Bat)

Ok, everything checks out... now we'll run the BBS.COM program from the PRO> directory.

If this is the first time you have run the board, several things must be done before you're ready for the first caller. First, you need to run the user editor (Option 6) so you can edit your ID (userID #10) in the userlog. Change the textual data to your liking and save the changes. Also edit the co-sysop user ID #11, even if you don't plan on having a co-sysop. The board treats ID's #10 and #11 differently then other ID's, so these must never be used by regular users.

User ID's 1 through 9 are reserved by the system. They are:
ID #1 is the 'Visitor record'
ID #2 is the 'New User record'
ID's 3 thru 9 are sysop predefined validation masks

Now, you will need to edit the Visitor record and the New User record to set up the following:

time limits
download ratios
command security levels
message base security levels
file sig security levels

When a new user calls and asks for a permanent ID, these are the initial values that will be transferred to their ID until you validate them. The rest of data is obtained from the user when they apply for an ID.

We recommend that VISITORS not be allowed to post messages. Just let them read messages and look around.

The ID's 3 thru 9 are special validation mask ID's (1-7) that the sysop can set up to aid in the validation of new users. Many of you currently running boards tend to validate new users in one of a handful of ways, depending on the user information supplied. You can predefine up to 7 of these validation masks so that when you are validating a new user, you can apply one of these validation masks to that user's ID. By doing this, all the command, message and file security levels along with the time limits and download ratio set in that particular validation mask are transposed to the user record that you are currently editing.

Rather than individually setting all the security flags by hand, you can APPLY a generic security level for this user by pressing just one key.

You will notice when the BBS starts, the 6 line Status Window at the top of the screen. This is the normal configuration. From time to time you may want or need to see more of the screen without all the status lines. By holding down the SHIFT CONTROL keys and pressing '1', '2', '3' or '4', you can change the number of lines used for the status window.

```
SHIFT CONTROL 1 - Displays the normal 6 line status window
SHIFT CONTROL 2 - Displays a 4 line status window
SHIFT CONTROL 3 - Displays a 2 line status window
SHIFT CONTROL 4 - No status window
SHIFT CONTROL 5 - Turns the screen OFF, increasing BBS
                  performance by 30%
SHIFT CONTROL 0 - Toggles high speed screen handler On/Off
SHIFT CONTROL D - Toggles the delete flag on the user online
SHIFT CONTROL H - Prompts the user they are over there time
                  and resets the BBS
SHIFT CONTROL T - Resets the online users time to 0 used in
                  the event they are close to running out of
                  time and are still reading messages
```

Sysop Options From Waiting for Call

There are 8 options available to the sysop while the board is waiting for a call. They are:

1 Quick Entry #1
2 Quick Entry #2
3 Logon By Handle
4 Logon By Name
5 Normal Logon
6 User Editor
7 Chat Mode
8 DOS Shell

1. Quick Entry #1 - this logon option allows the sysop (User #10) to quick logon directly to the command prompt. Logon using this option sets the last caller to 'A Visitor'.

2. Quick Entry #2 - this logon option allows a co-sysop (user #11) to quick logon directly to the command prompt. Logon using this option sets the last caller to 'A Visitor'.

3. Logon by Handle - this logon option allows logon directly to the command prompt as a user after supplying the users handle. Logon using this option updates the last caller.

4. Logon By Name - this logon option allows logon directly to the command prompt as a user after supplying the users real name. Logon using this option updates the last caller.

5. Normal Logon - this option allows logon as if you were connected remotely. All the normal logon files are displayed and all prompts for logon data are and validated.

6. User Editor - this option allows you to access the user editor without the need to logon to the system.

7. Chat Mode - this option toggles the chat mode ON and OFF. Chat mode status is visible in the BBS status window.

8. DOS Shell - this option allows you to access the DOS Shell without the need to logon to the system.

Entering Chat Mode

While a caller is online, you may enter chat mode by pressing the escape key once. You are then placed in a direct chat mode with the user. Typing from the local keyboard will immediately be output to the modem as you type. A unique feature of PRO's chat mode is if you press the escape key a second time, a type ahead buffer will appear in the bottom 2 status lines. You can now type a message to the user while they are typing to you, but your message will not be sent out until you press the enter key. Once you press the enter key, you are immediately back in real time chat mode. Pressing escape again will turn on the type ahead buffer again for your next message. This feature can came in handy if you are in chat mode with a slow typist and you know the answer you want to send before the caller has even finished asking the question. Just press escape again, key your response and wait for them to finish typing before pressing the enter key to send your response. Your users will be amazed at how fast a typist you are!

To EXIT chat mode, simply press ESC twice.

The System Clock

The BBS system clock is actually using the SpartaDOS software clock which is normally set manually or automatically set if you have the R-Time 8 clock cartridge. In either case, BBS Express! PRO requires that TDLINE.COM handler be installed before running the BBS. An error will occur and PRO will refuse to load if the TDLINE handler is not installed.

The Call Log

The system call log is stored in the PRO>LOG> subdirectory using the name CALL.LOG. Each caller to the BBS is logged into this dataset. Call.Log does not have to be predefined. If PRO can not find it, the dataset will automatically be created.

The Call.Log may be disabled if you choose, by setting the LOG drive number to 0 using the SysEditor.

User Editable Parameters

When a user presses the P key from the main command prompt to edit their user settable parameters, they are presented with the following:

```
[A] Real Name: John Doe
[B]    Handle: JD
[C]   Address: 1234 Any Street
[D]      City: Augusta, Ga
[E]  Zip Code: 23113
[F]   Country: USA
[G]     Phone: 379-4156
[H]  Computer: Atari (all)
[I]       Age: 30
[J]  Password: atari
[K]  V. Width: 40
[L] Auto-read E-mail at logon: No
[M] Clear screen after msg  : Yes
[N] "Hotkey" 1-key commands : Yes
[O] Non-stop quickscan mode : No
[P] Message view page length: 0

Enter letter to change, OK when done,
or <return> to list:
```

Any of these settings may be set by the user (except changing their name/handle) and are stored in the users record. PRO uses these settings when making decisions on how to present the data to the user.

Logging A User Off The System

At any time a caller is online, you may hang them up instantly by pressing the START SELECT OPTION keys simultaneously. This will terminate the call and reset the board back to the waiting for call screen.

Printing and Purging The User Log

There are 2 .CMD modules available on the disk for the purpose of printing and purging the userlog. They are: ULPRINT.CMD and ULPURGE.CMD

ULPRINT.CMD is a command module that can only be run from the Dos Shell by typing ULPRINT. It has 5 options available:

1. Print Userlog To Printer - will print the userlog (4 users per page) to the printer.

2. Print Userlog To Disk File - will write the userlog report to a disk file instead of the printer. You will be prompted for a FULL filename at the time you execute this function. This can be useful for remote SysOps. They can download the created file and copy it to their printer or for the sysop to print the userlog on another system that has a printer connected. The output from this option is the same as in option 1 except it goes to a disk file rather than the printer.

3. Print Only New Users - will print only the users in the userlog who currently have a status of 'New User'. This can be useful for printing new users since the last time you validated users, BEFORE validating them. Output from this option is directed to the printer.

4. Print Visitor/NewUser/Masks - will print the visitor, new user and 7 validation masks from the userlog. This is a handy way to review these records a have a permanent printout available when validating new users.

5. View Userlog Statistics - will scan the userlog and display to the screen (locally and remotely) the current userlog statistics as follows:

```
System Reserved : x
Active Users    : xxxxx
Locked Users    : xxxxx
New Users       : xxxxx
Deleted Users   : xxxxx
Total Users     : xxxxx
```

This can be useful for determining the status of your userlog and to see if new users have called since your last validation or if the userlog is approaching its maximum allocation.

ULPURGE.CMD is a command module that can be run from the Dos Shell by typing ULPURGE, or it may be set up as an event to be run by the Event Scheduler.

When ULPURGE runs, it looks for a file called ULPURGE.DAT in the PRO>DATA> subdirectory. This file is a text file containing 1 line. On that line is a number between 1 and 365 which represents the number of days back from the current date to start purging.

Let's say you want to delete users who have not called your board in the last 90 days. Create the file called ULPURGE.DAT in the PRO>DATA> subdirectory. On the first line, key in 90, followed by a return, and save the file. Now when ULPURGE.CMD runs, it will read the ULPURGE.DAT and use 90 as its days to purge value. It will read the userlog and mark any user who has not called in the last 90 days as deleted. ULPURGE will bypass any user record you have locked regardless of the last call date, so locked users will never be deleted.

The important thing to remember here is that once you establish your purge days, you never have to change. ULPURGE will always calculate the purge date based on the current date and your purge days criteria. While you can run this program as often as every call, we suggest you use it as an event and set it up to run once a week.

Other Utility Programs

ULBACKUP.CMD is a command module that can be run from the Dos Shell by typing ULBACKUP while in the Dos Shell or it may be set up as event to be run by the event Scheduler. When ULBACKUP runs, it looks for a file called ULBACKUP.DAT in the PRO>DATA> subdirectory. This file is a text file containing 1 line. On that line is the FULL filename to be used as the destination filename for backing up the userlog and is terminated by a return. We suggest you set up your filename to backup to the same directory as your userlog using the name USERLOG.BAK. So, assuming your userlog is on drive 2, ULBACKUP.DAT would contain one line with the filename D2:>PRO>USERLOG>USERLOG.BAK followed by a return. How often you run the ULBACKUP is a function of where you have your userlog located. If it's on the hard drive or a floppy drive, once a day or once a week is probably fine. If you run the userlog out of a ramdisk, you might consider setting ULBACKUP to run after every call. Then if you lose power, you really haven't lost anything. When the power comes back on, recopy the userlog back to the ramdisk and you're back in business.

WHOMAINT.CMD is a command module that records the caller to the last 50 callers log which users can view using the WHOCALL.CMD module. By default, this module was automatically set up when you configured your sysdata.dat file to be the first event to run and will always run after every call. To insure the accuracy of your lastcall data file, we suggest you run it like this unless you do not wish to use the last call function on the board, in which case this module may be removed from the event scheduler. WHOMAINT will not record the logon of user number 10 (Sysop) or user number 11 (Co-Sysop) when they logon either locally or remotely. This is so you or your co-sysop may enter your BBS without the users being aware of the fact. The actual data filename is LASTCALL.DAT and is stored in the PRO>DATA> subdirectory. This file does not need to pre-exist. WHOMAINT will create a new file if it can't find one. Should you ever need or want to, LASTCALL.DAT may be deleted without any ill effect. WHOMAINT will simply create a new one the next time it runs.

D.CMD is a command module which is an alternate to the Dos Shell's internal DIR command. D will list the files 2 to a line instead of the usual 1 per line.

EDITFILE.CMD is a command module which is used to edit text files while the BBS is running. EDITFILE is capable of editing a 255 line text file of up to 8 columns. EDITFILE is run from the Dos Shell by entering the module name. You may pass the filename as a parameter to EDITFILE or just be prompted for a filename by entering the module name.

VDEL.CMD is a command module which will ask for verification before deleting the file. It may be used as an alternate to the Dos Shell's internal ERASE a DELETE commands which do not ask for verification before deleting the file.

WHEREIS.CMD is a command module to be run from the Dos Shell which will locate a file or files based on the input filemask any where on the drive. For instance, keying a filemask D3:>*.ARC would locate all files on drive 3 with an extender of ARC regardless of which subdirectory they were in.

VIEWEVNT.CMD is a command module to be run from the Dosshell which will display the last run status of the event scheduler. It will list each event and last date/time it was run with the a return code run status. A run status of 0 indicates the event ran successfully while a non zero return code indicates an error occurred during execution. The error code will be a DOS return code. For example, a return code of 170 would indicate the event was looking for a file which it could not find.

UEXTEND.COM is a stand alone program to be used for extending your userlog. No matter how well you plan, the time may come when you run out of ID's in your userlog. UEXTEND will increase the current userlog by an additional number of entries that you specify when running the program. This program modifies the original userlog, so only 1 drive

is required to perform this operation. But, make sure you have a backup of the userlog before running this program. If something should happen like the drive fills up while the program is running, you can recover by copying your backup copy to the original userlog and try again. Run this program from the DOS prompt and follow the program prompts. This program may not be run while the BBS is running.

LOGON.CMD eliminates the need to chain files to each other with the creation of a LOGON.DAT file which will be placed in your >DATA> subdirectory.

LOGON.DAT FILE use editfile to create or any text processor example:

```
-B
||
|└──────── B represents Steve's Box if you have it.  If not type a -
└───────── This - is for Random Screens type R to turn them on.
           Note: Random screens, to have them you must create, CONNECT.ATA
           and .ASC or with the other Help subdirectories Connect.HLP
           files CONNECTA-F and CONNECT0-9.  You may create 1 or 2 screens
           and copy them over to get it to work.

SYSNEWS H O
|       | |
|       | └──────────────── O is Security level of user to View file
|       └────────────────── H tells it to go to Help File subdirectory
└──────────────────────────── Sysnews is the command filename
```

```
Example: LOGON.DAT
R-
SYSNEWS H O
QUICKSCAN CQ 1
READMAIL C 1
D6:>PRO>SMILE.TXT D 31
```

In the example above the Random Screens are on the Box is off. The user will automatically be shown sysnews (level user 0), Quickscan will be shown to the user if they have a security level of 1, readmail same, D6: this provision allows the sysop to dump any file to the screen to be viewed by all users with the security level set.

Commands that can be used in place of the D=dump are C=command, T=text, H=Help and the security level can be set anywhere from 0-32. Quickscan CQ 1. The Q means Invoke a single byte parameter that has to be passed to command file being called. Quickscan is the only command file that requires this in PRO.

MENU.CMD External Menu Processor

MENU.CMD is a multi-purpose menu processor that allows you to mix both command and text files in the same menu. Up to 99 entries are allowed per menu. In addition, MENU.CMD allows you to move .CMD files out of your >COMMANDS> subdirectory. One other advantage in using this external menuing system is that, unlike the internal menuing system, once a command is executed or a text file is displayed, the menu will regain control and redisplay itself to the user for the next selection.

MENU.CMD setup can be a little tricky, so get you a fresh cup of coffee and pay close attention. One of the limitations of specifying commands in the SYSDATA "main command table" is that there is no way for you to specify a "command line" with arguments to pass to the program you are about to execute.

One of the main reasons we wrote MENU.CMD was to allow .CMD files to be placed in subdirectories other than the >COMMANDS> subdirectory. So, how does MENU.CMD know which subdirectory to use? Well, it gets its info from the command name itself. You simply rename (or copy) MENU.CMD to M_xxxxxx.CMD, where the 'xxxxxx' is the subdirectory name to be used for this menu. These 'xxxxxx' subdirectories are found in your system >DATA> subdirectory.

For example, if you were going to set up 3 menu processors named GAMES, LIBRY, and UTILS (for games, library, and utilities) you would do the following:

- make 3 copies of MENU.CMD named:

  M_GAMES.CMD
  M_LIBRY.CMD
  M_UTILS.CMD

- create 3 subdirectories under your >DATA> subdirectory named:

  Dx:>PRO>DATA>GAMES>
  Dx:>PRO>DATA>LIBRY>
  Dx:>PRO>DATA>UTILS>

- create 3 menu files in those subdirectories named:

  Dx:>PRO>DATA>GAMES>GAMES.MNU
  Dx:>PRO>DATA>LIBRY>LIBRY.MNU
  Dx:>PRO>DATA>UTILS>UTILS.MNU

The .MNU file for a menu contains three pieces of information for each entry in a menu: (1) whether this entry is a "T"ext or "C"ommand file, (2) the name the file to view/execute, and (3) the description to be shown to the user.

THERE MUST BE NO BLANK LINES IN THE .MNU FILE!

The layout is as follows:

```
  The Title of the Menu            ----> first line is the "menu title"
c filename This is the description ----> signals a "c"ommand file
t filename This is the description ----> signals a "t"ext file
c filename This is the description ----> signals a "c"ommand file
t filename This is the description ----> signals a "t"ext file
p filename This is the description ----> a command file with
                                        "P"arameters
```

The "filename" MUST BE 8 CHARACTERS LONG AT ALL TIMES. If the filename is shorter than eight characters, then fill it out with spaces. All of the .CMD files get executed from the menu subdirectory, not from the >COMMANDS> subdirectory. Also, all text files get viewed from the menu subdirectory. If the user is in 80 column mode, the menu program looks for a text file with an extension of ".80". If they are in 40 column mode, it looks for a file with an extension if ".40". In addition, if there is no ".80" file there, then the menu processor will go ahead and view the ".40" file. This means you can simply keep only the 40-column versions of text files if you wish.

As an example, we will now go through the steps to set up a menu process. A good example will be to create a "GAMES" menu where you can place all of your game .CMD files. Here are the steps you would perform to do this (we will assume that you have 3 games named KINGDOM.CMD, CHESS.CMD, and HANGMAN.CMD).

1.Copy MENU.CMD to M_GAMES.CMD.

2.Install "M_GAMES" in your "main command" table in the SYSDATA file.

3.Create a directory called >GAMES> in your system >DATA> directory.

4.Create a text file named GAMES.MNU in your new >GAMES> directory with the following data (each line beginning in column 1):

```
BBS Pro Game Menu
C kingdom  The Kingdom Game
C chess    The Chess Manager
C hangman  Hangman; Guess the words!
```

5. Copy Dx:>pro>commands>kingdom.cmd to Dx:>pro>data>games>kingdom.cmd

6. Copy Dx:>pro>commands>chess.cmd to Dx:>pro>data>games>chess.cmd

7. Copy Dx:>pro>commands>hangman.cmd to Dx:>pro>data>games>hangman.cmd

8. Delete Dx:>pro>commands>kingdom.cmd

9. Delete Dx:>pro>commands>chess.cmd

10. Delete Dx:>pro>commands>hangman.cmd

That's all there is to it! MENU.CMD provides a very nice display to the user with the ability to page forward and backward throughout the menu. But, its plus is that it allows you to ease up the tension of the >COMMANDS> subdirectory filling up. Use your imagination with MENU; it can do many things. Here are some things you could do:

o A library full of text files.

o A "sysop utility" menu with your most-used sysop commands.

o A "file menu" that replaces the "B"rowse command from the main command. Could take them to a menu with options like:

o Display file stats (filestat)

o Search download files (findfile)

o Enter the file processor (browse)

Setting up ANSI Screens

All the "&xx" are still valid. It is now possible to do all your IBM screens on your IBM using ANSI Drawing programs. I use a Shareware program called ANSI DRAW. Edit your ANSI Screen, save it to disk without a Clear Screen Code. Send it up to your BBS. Run TOATARI.CMD on it. Boot up TextPro and insert a &00, then call the file into textpro. Strip any junk off the end of the file. Now for some strange reason MS Dos sometimes adds junk to the end of a file or your terminal program does. When you look at these ANSI Screens you will see all kinds of, ansi codes, don't worry you will not see the codes when your BBS is up and running. If you are going to use ansi codes within a COMMAND_MENU or TEXT_MENU make your ANSI Screen without the menu, and add the menu structure after you move it bak to the 8-bit. Remember your COMMAND_MENUS and TEXT_MENUS can not have ANSI CODES within their menu structure.

Example: This portion can not have ANSI CONTROL codes within it.

```
COMMAND_MENU LEVEL 20
Play which game?------>
1 BLACKJACK
2 KING
exit/exit
end
&00
```

```
_______________This below can have ansi control codes___________

   [1] Play Black Jack                 [2] Play Kingdom

                     Exit=Exit etc..
```